I claim:

1. A method of configuring IA-32 computer resources, comprising:

determining an amount of physical memory available and a minimum memory-mapped IO size;

if the amount of physical memory available is less than or equal to 4GB:

determining a rounded memory-mapped IO size by rounding up the minimum memory-mapped IO size to the next multiple of 128M;

else:

determining a number of DIMM socket pairs available; and

determining the rounded memory-mapped IO size by rounding up the minimum memory-mapped IO size to the next multiple of x, where x is a function of the number of DIMM socket pairs available; and

setting the top of lower memory equal to 4GB minus the rounded memory-mapped IO size.

2. The method of claim 1, wherein:

x equals 256M if the number of DIMM socket pairs available equals 2.

3. The method of claim 1, wherein:

x equals 512M if the number of DIMM socket pairs available equals 3.

4. The method of claim 1, wherein:

x equals 1024M if the number of DIMM socket pairs available equals 4.

5. The method of claim 1, wherein:

x equals 2G if the number of DIMM socket pairs available exceeds 4.

6. The method of claim 1, further comprising:

programming one or more pairs of MTRR's to define caching characteristics of the physical memory without including the PCI memory address range in any region defined by the one or more pairs of MTRR's.

7. The method of claim 1, further comprising:

using a maximum of 6 pairs of MTRR's to define caching characteristics of the physical memory.

8. The method of claim 6, further comprising:

using a maximum of 6 pairs of MTRR's to define the caching characteristics of the physical memory.

9. A machine-readable storage or transmission medium containing code that, when executed on a computer, causes the computer to perform a method of configuring IA-32 computer resources, the method comprising:

determining an amount of physical memory available and a minimum memory-mapped IO size;

if the amount of physical memory available is less than or equal to 4GB:

determining a rounded memory-mapped IO size by rounding up the minimum memory-mapped IO size to the next multiple of 128M;

else:

determining a number of DIMM socket pairs available; and

determining the rounded memory-mapped IO size by rounding up the minimum memory-mapped IO size to the next multiple of x, where x is a function of the number of DIMM socket pairs available; and

setting the top of lower memory equal to 4GB minus the rounded memory-mapped IO size.

10. The storage or transmission medium of claim 9, wherein:

x equals 256M if the number of DIMM socket pairs available equals 2.

11. The storage or transmission medium of claim 9, wherein:

x equals 512M if the number of DIMM socket pairs available equals 3.

12. The storage or transmission medium of claim 9, wherein:

x equals 1024M if the number of DIMM socket pairs available equals 4.

13. The storage or transmission medium of claim 9, wherein:

x equals 2G if the number of DIMM socket pairs available exceeds 4.

14. The storage or transmission medium of claim 9, wherein the method further comprises:

programming one or more pairs of MTRR's to define caching characteristics of the physical memory without including the PCI memory address range in any region defined by the one or more pairs of MTRR's.

15. The storage or transmission medium of claim 9, wherein the method further comprises:

using a maximum of 6 pairs of MTRR's to define caching characteristics of the physical memory.

16. The storage or transmission medium of claim 14, wherein the method further comprises:

using a maximum of 6 pairs of MTRR's to define the caching characteristics of the physical memory.